Yee & Associates, P.C.

4100 Alpha Road
Suite 1100
Dallas, Texas 75244

Main No. (972) 385-8777
Facsimile (972) 385-7766

RECEIVED
CENTRAL FAX CENTER
MAR 1 8 2005

Facsimile Cover Sheet

To: Commissioner for Patents for **Examiner Jason D. Cardone** **Group Art Unit 2145**	Facsimile No.: **703/872-9306**
From: Jane M. Roberts for Michele Morrow Legal Assistant to Gerald H. Glanzman	No. of Pages Including Cover Sheet: **22**

Message:

Enclosed herewith:

- Transmittal Document; and
- Appeal Brief.

Re: Application No. **09/976,126**
Attorney Docket No: **RSW919980041US2**

Date: Friday, March 18, 2005

Please contact us at (972)385-8777 if you do not receive all pages indicated above or experience any difficulty in receiving this facsimile.	*This Facsimile is intended only for the use of the addressee and, if the addressee is a client or their agent, contains privileged and confidential information. If you are not the intended recipient of this facsimile, you have received this facsimile inadvertently and in error. Any review, dissemination, distribution, or copying is strictly prohibited. If you received this facsimile in error, please notify us by telephone and return the facsimile to us immediately.*

PLEASE CONFIRM RECEIPT OF THIS TRANSMISSION BY FAXING A CONFIRMATION TO 972-385-7766.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Dingsor et al.**	§	Group Art Unit: **2145**
Serial No.: **09/976,126**	§	Examiner: **Jason D. Cardone**
Filed: **October 11, 2001**	§	Attorney Docket No.: **RSW919980041US2**
For: **Method and Apparatus Load Balancing Server Daemons within a Server**	§	

Certificate of Transmission Under 37 C.F.R. § 1.8(a)
I hereby certify this correspondence is being transmitted via facsimile to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, facsimile number (703) 872-9306 on March 18, 2005.

By: Jane M. Roberts
Jane M. Roberts

36736
PATENT TRADEMARK OFFICE
CUSTOMER NUMBER

TRANSMITTAL DOCUMENT

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:
ENCLOSED HEREWITH:

- Appeal Brief (37 C.F.R. 41.37).

A fee of $500.00 is required for filing an Appeal Brief. Please charge this fee to IBM Corporation Deposit Account No. 09-0461. No additional fees are believed to be necessary. If, however, any additional fees are required, I authorize the Commissioner to charge these fees which may be required to IBM Corporation Deposit Account No. 09-0461. No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and I authorize the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0461.

Respectfully submitted,

Gerald H. Glanzman
Registration No. 25,035
Duke W. Yee
Registration No. 34,285
YEE & ASSOCIATES, P.C.
P.O. Box 802333
Dallas, Texas 75380
(972) 385-8777
ATTORNEYS FOR APPLICANTS

Docket No. RSW919980041US2 *PATENT*

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

RECEIVED
CENTRAL FAX CENTER
MAR 1 8 2005

In re application of: **Dingsor et al.**	§	
	§	Group Art Unit: **2145**
Serial No. **09/976,126**	§	
	§	Examiner: **Cardone, Jason D.**
Filed: **October 11, 2001**	§	
	§	
For: **Method and Apparatus Load Balancing Server Daemons within a Server**	§ §	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Certificate of Transmission Under 37 C.F.R. § 1.8(a)
I hereby certify this correspondence is being transmitted via facsimile to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, facsimile number (703) 872-9306 on March 18, 2005.

By: Jane M Roberts
Jane M. Roberts

APPEAL BRIEF (37 C.F.R. 41.37)

This brief is in furtherance of the Notice of Appeal, filed in this case on February 1, 2005.

The fees required under § 41.20(B)(2), and any required petition for extension of time for filing this brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

REAL PARTY IN INTEREST

The real party in interest in this appeal is the following party: International Business Machines Corporation.

RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

STATUS OF CLAIMS

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1-41

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims canceled: 1-15 and 21-37
2. Claims withdrawn from consideration but not canceled: NONE
3. Claims pending: 16-20 and 38-41
4. Claims allowed: 18 and 39-41
5. Claims rejected: 16, 17, 19, 20 and 38
6. Claims objected to: NONE

C. CLAIMS ON APPEAL

The claims on appeal are: 16, 17, 19, 20 and 38

STATUS OF AMENDMENTS

An Amendment after Final Rejection was not filed. Therefore, claims 16, 17, 19, 20 and 38 on appeal herein are as presented in the Response to Office Action filed July 28, 2004.

SUMMARY OF CLAIMED SUBJECT MATTER

A. CLAIM 16 - INDEPENDENT

The subject matter of claim 16 is directed to a computer. The computer **104** comprises a plurality of processes (see **Figure 3**, Specification, pg. 11, lines 14-27), wherein the plurality of processes service a destination address and have process addresses (see **Figure 3**, Specification, pg. 12, lines 7-26), a packet routing layer **404**, wherein packet routing layer **404** routes packets to the plurality of processes using a destination addresses within the packets (see **Figure 4**, Specification, pg. 13, lines 13-19), and a dispatch layer **406** between a TCP layer and an IP layer (see **Figure 4**; Specification, pg. 13, line 20 through pg. 14, line 5). Dispatch layer **406** has a plurality of modes of operation. These modes include a first mode of operation in which dispatch layer **406** receives packet **402** from a client, wherein packet **402** includes destination address **549** (see **Figure 8**, Specification, pg. 17, line 26 through pg. 18, line 1). In a second mode of operation, responsive to receiving packet **402**, dispatch layer **406** identifies a process within the plurality of processes to service the client, wherein the process is an identified process (see **Figure 8**, Specification, pg. 18, lines 14-20 and pg. 19, lines 21-25). In a third mode of operation, dispatch layer **406** translates the destination address **549** to a process address for the identified process within the plurality of processes (see **Figure 8**, Specification, pg. 19, lines 11-17). In a fourth mode of operation, which is responsive to the third mode of operation, packet **402** is sent to packet routing layer **404** (see **Figure 8**, Specification, pg. 19, lines 11-17).

B. CLAIM 38 - INDEPENDENT

Claim 38 is a computer program product claim counterpart to claim 16 and recites similar subject matter to that recited in claim 16.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

A. GROUND OF REJECTION 1 (Claims 16, 17, 19 and 38)

Claims 16, 17, 19 and 38 stand rejected under 35 U.S.C. § 102(e) as being anticipated by Brendel (U.S. Patent No. 5,774,660).

B. GROUND OF REJECTION 2 (Claim 20)

Claim 20 stands rejected under 35 U.S.C. § 103(a) as being obvious over Brendel in view of Coile et al. (U.S. Patent No. 6,061,349).

ARGUMENT

A. 35 U.S.C. § 102(e), Anticipation, Claims 16, 17, 19 and 38

The examiner has rejected claims 16, 17, 19 and 38 under 35 U.S.C. § 102(e) as being allegedly anticipated by Brendel et al. (U.S. Patent No. 5,774,660).

As to independent claims 16 and 38, the Final Office Action states:

> 4. Regarding claims 16 and 38, Brendel discloses a computer comprising:
> a plurality of processes, wherein the plurality of processes service a destination address and have process addresses [Brendel, col. 13, lines 18-46];
> a packet routing layer, wherein the packet routing layer routes a packets to the plurality to the plurality of processes using a destination addresses within the packets [ie. TCP layer, Brendel, col. 13, lines 18-46];
> a dispatch layer between a TCP layer and an IP layer, wherein the dispatch layer has a plurality of modes of operation including: a first mode of operation in which the dispatch layer receives a packet from a client, wherein the packet includes the destination address [ie. raw socket, Brendel, col. 14, line 56 – col. 15, line 56];
> a second mode of operation, responsive to receiving the packet, in which the dispatch layer identifies a process within the plurality of processes to service the client, wherein the process is an identified process [Brendel, col. 14, line 41-col. 15, line 16];
> a third mode of operation in which the dispatch layer translates the destination address to a process address for the identified process within the plurality of processes;
> and a fourth mode of operation, responsive to the third mode of operation, in which the packet is sent to the packet routing layer [Brendel, col. 17, lines 9-26].

Final Office Action dated December 1, 2004, pages 2-3.

The examiner goes on to state:

> (A) Brendel does not disclose a dispatch layer between a TCP layer and an IP layer that has four modes of operation.
> As to point (A), Brendel does disclose a modified IP input module and raw socket that has the four modes of operation of the instant independent claims 16 and 38 [Brendel, col. 15, lines 11-56]. During patent examination and prosecution, claims must be given their broadest reasonable interpretation. *In re*

> *Van Geuns*, 988 F.2d 1181, 1184, 26 SPQ2d 1057, 1059 (Fed. Cir. 1993); *In re Prater*, 415 F.2d 1393, 1404, 162 USPQ 541,550 (CCPA 1969). Giving the instant claims their broadest reasonable interpretation "a dispatch layer between a TCP layer and an IP layer" is broad enough to read on the modified IP input module and raw socket that are between a TCP layer and an IP layer disclosed in Brendel.

Final Office Action dated December 1, 2004, pages 4-5.

Independent claim 16 recites:

> 16. A computer comprising:
>
> a plurality of processes, wherein the plurality of processes service a destination address and have process addresses;
>
> a packet routing layer, wherein the packet routing layer routes packets to the plurality of processes using a destination addresses within the packets;
>
> <u>a dispatch layer between a TCP layer and an IP layer, wherein the dispatch layer has a plurality of modes of operation including:</u>
>
> a first mode of operation in which the dispatch layer receives a packet from a client, wherein the packet includes the destination address;
>
> a second mode of operation, responsive to receiving the packet, in which the dispatch layer identifies a process within the plurality of processes to service the client, wherein the process is an identified process;
>
> a third mode of operation in which the dispatch layer translates the destination address to a process address for the identified process within the plurality of processes; and
>
> a fourth mode of operation, responsive to the third mode of operation, in which the packet is sent to the packet routing layer.

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). Appellants respectfully submit that Brendel does not identically show every element of the claimed invention arranged as they are in the claims. Specifically, Brendel does not teach a dispatch layer between a TCP layer and an IP layer or such a dispatch layer that has the four modes

of operation recited in claim 16.

Brendel is directed to a system for resource-based load balancing on a distributed resource multi-node network. With the system of Brendel, a load balancer receives all requests from clients using virtual address for a web site. The load balancer makes a connection with the client and waits for a URL of the resource requested by the client. The load balancer waits to perform load balancing until after the location of the requested resource is known. The connection and URL request are passed from the load balancer to a second node having the requested resource. The load balancer re-plays the initial connection packet sequence to the second node but modifies the address to that of the second node. The network software is modified to generate the physical network address of the second node but then changes the destination address back to the virtual address. Since all requests are first received by the load balancer which determines the physical location of the requested resource, nodes may contain different resources. The entire contents of the web site are not mirrored onto all nodes. Network bottlenecks are avoided since the nodes transmit files back to the client directly, bypassing the load balancer.

Thus, Brendel teaches a load balancing mechanism in which a load balancer identifies the location of a requested resource and then modifies the virtual address received from the client to an address for the node having the requested resource. The node may then use the virtual address to directly communicate the requested resource to the client without having to go through the load balancer. Brendel does not teach a dispatch layer that is between a TCP layer and an IP layer. Furthermore, Brendel does not teach a dispatch layer that is between a TCP layer and an IP layer and that has the four modes of operation recited in claim 16 of the present application.

Column 15, lines 36 through 38 of Brendel teaches that:

> The invention performs additional steps before step **306** by modifying the generic IP input module to form modified IP input module **200**.

Column 14, lines 41 through 43 of Brendel teaches that:

> Unmodified link layer **84** passes packets received up to TCP/IP stack **82**, and specifically to IP input module 200 of the IP layer.

(emphasis added)

The above cited two passages of Brendel teach that the modified IP input module is made by modifying a normal, basic IP input module. As Brendel specifically teaches that an IP input module is part of the IP layer, it follows that a modified IP module is also part of the IP layer and not a separate "dispatch layer between a TCP layer and an IP layer," as recited in claim 16. Therefore, the examiner's assertion that "[g]iving the instant claims their broadest reasonable interpretation "a dispatch layer between a TCP layer and an IP layer" is broad enough to read on the modified IP input module and raw socket that are between a TCP layer and an IP layer disclosed in Brendel," is incorrect since the modified IP input module and raw socket are not between the TCP layer and the IP layer. Instead, the modified IP input module is part of the IP layer. Also, as taught in column 14, line 59 through 61, of Brendel, cited below, raw sockets are standard features of TCP/IP, and therefore part of TCP/IP.

The Final Office Action alleges that Brendel teaches a dispatch layer that is between a TCP layer and an IP layer at column 14, line 56 to column 15, line 56 which reads as follows:

> Local packets that are not of a known protocol such as TCP or UDP (User Datagram Protocol) have an unrecognized protocol. These datagrams are sent to raw socket **214**, bypassing TCP module **218**. Any applications in application layer **80** can listen to raw socket **214** and use the datagram, since raw sockets are a standard TCP/IP feature. Load balancer 70 is an application which listens to raw socket **214** for datagrams using the "IXP" protocol. Since the IXP protocol is not a defined protocol, no other applications should be looking for IXP datagrams. Thus using the IXP protocol allows use of raw socket **214** to bypass the TCP layer and send the datagrams directly to load balancer **70**. These datagrams are the connection packets and the URL originally from the client's browser.
>
> Each server is modified to accept packets using the virtual IP address by aliasing a second IP address, thus using two IP addresses. For example, in UNIX, the command:
>
> % ifconfig de0 230.101.17.200 alias netmask Oxffffffff
>
> specifies that a second IP address, the virtual IP address 230.101.17.200 is also an IP address for the node. Other operating systems also support IP address aliasing.
>
> Modified IP Input Module--FIG. **15**
>
> FIG. **15** is a flowchart for a modified IP layer input module. The server with the load balancer uses <u>modified IP input module **200**</u>. An asterisk is used to indicate that the module is modified from the generic ip_ input() module. Steps

308, **310**, **312**, and **314** are added steps which are not in the generic IP module.

All packets received from the media by the lower link layer are passed up to the IP layer which calls IP input module **200**. Step **302** tests to determine if the packet is for the local node by reading the destination IP address.

When step **302** determines that the destination IP address is not a local IP address, then the packet is being routed through the local node and the IP layer acts as a software router. The packet is passed to IP forward module **202** (step **304**) which prepares the packet for forwarding. The packet is then sent to IP output module **206** before being re-transmitted out the link layer to the destination or the next hop.

Step **302** determines that the packet is for the local node when the IP address is the virtual IP address or the real IP address for the server. The packet is stripped of its header information and possibly assembled with other packets to form the IP datagram, step 306.

The assembled IP datagram from step 306 is normally sent up to the TCP layer (steps **316**, **318**) for the generic IP module. The invention performs additional steps before step **306** by modifying the generic IP input module to form modified IP input module **200**. Modified IP input module **200** checks the protocol to determine if it is the IXP protocol. Since incoming packets from the Internet always use the TCP protocol, incoming packets fail step **308** and are then tested by step 310 to determine if they are TCP packets with the virtual IP address and are world-wide-web packets. Thus step 310 looks for incoming packets. These incoming packets have their protocols changed from TCP to IXP, step **314**. The IXP protocol is not a recognized protocol, so step **316** causes these incoming packets to be sent to the raw socket, step **320**, so that the load balancer application can read these packets. Thus changing the protocol to the unrecognized IXP protocol forces the incoming packets to be sent directly to the load balancer. This allows all incoming packets from the Internet to be routed through the load balancer.

Other TCP packets which are not world-wide web packets fail step **310** and are not modified. These ordinary TCP packets are a known protocol, step **316**, and are sent to the TCP layer. step **318**.

(emphasis added)

The most pertinent part of the above cited portion of Brendel is the description of Figure 15. This description states that the IP layer calls the modified IP module 200. The modified IP module 200 checks to see if a received data packet is in the IXP protocol and if not, converts the data packet from TCP to IXP so that the data packet is directly sent to a raw socket for reading by the load balancer. There is no mention of any dispatch layer between a TCP layer and an IP layer in this, or any other, section of Brendel. While the TCP layer and the IP layer are mentioned, the

actual functions of converting protocols from TCP to IXP is performed in the IP layer by calling a modified IP module 200. Furthermore, the functions that are performed by this modified IP module 200 merely converts protocols and does not perform the functions of the dispatch layer as recited in claim 16.

Claim 16 specifically recites that the dispatch layer has four modes of operation. A first mode of operation is one in which the dispatch layer receives a packet from a client, wherein the packet includes the destination address. A second mode of operation is one in which, responsive to receiving the packet, the dispatch layer identifies a process within the plurality of processes to service the client, wherein the process is an identified process. Nowhere in the cited section of Brendel is there any teaching of a dispatch layer that is between a TCP layer and an IP layer that performs the function of the second mode set forth in claim 16.

The Final Office Action alleges that this second mode of operation is taught by Brendel at column 14, line 41 to column 15, line 16 which is part of the reproduced section of Brendel above. However, nowhere in this section is there any teaching regarding a dispatch layer between a TCP layer and an IP layer, identifying a process within a plurality of processes to service a client. In fact, the section cited as allegedly teaching these features is under the heading "Modified IP layer – FIG. 14" (column 14, line 38). Furthermore, the section discusses determining if the packet is destined for a local node or another node and that local packets are assembled into IP datagrams. There is no teaching or suggestion regarding a dispatch layer between an IP layer and a TCP layer identifying a process within a plurality of processes to service a client.

A third mode of operation of the dispatch layer, as recited in claim 16, is one in which the dispatch layer translates the destination address to a process address for the identified process within the plurality of processes. The fourth mode of operation is one in which, responsive to the third mode of operation, the packet is sent to the packet routing layer. Again, there is no teaching or suggestion in Brendel with regard to a dispatch layer between a TCP layer and an IP layer performing such a function. The Final Office Action cites column 17, lines 9-26 as allegedly teaching the third and fourth modes of operation. However, the cited section does not teach either the features of the third mode of operation or the features of the fourth mode of operation.

Column 17, lines 9-26 reads as follows:

> Incoming packets which are assigned to the load balancer node's server are passed up and down the local TCP/IP stack twice. These packets are first sent from the low-level link layer through the modified IP layer to the load balancer in the application layer, and then back down through the IP layer to the link layer. Step **336** of FIG. **16** detects that the local server is the destination and bypasses steps **338**, **340** so that the protocol is left as IXP.
>
> The link layer recognizes that the NIC address is the local NIC address and does not transmit the packets. Instead the packets are sent back up to the IP layer. Step **308** of FIG. **15** detects these packets and changes the protocol back to TCP (step **312**) and then passes the TCP packets to the HTTPD server application through the generic TCP layer. This sequence only occurs for a packet that has been intercepted to the load balancer and assigned to the server on the local node.

This section of Brendel provides further support for the distinction of the present claims reciting a dispatch layer between an IP layer and a TCP layer, a feature not taught by Brendel. This section of Brendel teaches that the packet is sent from the link layer to the modified IP layer and then to the application layer bypassing the TCP layer. Thus, Brendel teaches a mechanism in which the IP layer is modified to bypass the TCP layer and send data packets directly to the load balancer in the application layer. Brendel does not teach or suggest a dispatch layer between a TCP layer and an IP layer.

Furthermore, Brendel does not teach a dispatch layer that performs the functions of operational modes three and four in claim 16. That is, there is no teaching in the above cited section regarding a dispatch layer between a TCP layer and an IP layer that translates a destination address to a process address for an identified process within a plurality of processes or that, responsive to this translation, sends the packet to a packet routing layer. While Brendel may teach modifying protocols from TCP to IXP and back again, there is nothing in Brendel that teaches a dispatch layer between a TCP layer and an IP layer that identifies a process from a plurality of processes to service a client, translates a destination address to a process address for the identified process, and then sends the data packet to a packet routing layer.

Independent claim 38 recites similar features to that of claim 16. Specifically, claim 38 recites instructions for translating, in a dispatch layer between a TCP layer and an IP layer, a destination address to an intermediate destination address which is an address for a selected process within a plurality of processes. Similar features to these have been addressed above with regard to claim 16 and thus, claim 38 is distinguished over the Brendel reference for similar

reasons. Brendel does not teach or suggest these features in claim 38.

In view of the above, Appellants respectfully submit that Brendel does not teach each and every feature of independent claims 16 and 38. At least by virtue of their dependency on claim 16, Brendel does not teach each and every feature of dependent claims 17 and 19. Accordingly, Appellants respectfully request that the Board reverse the examiner's Final Rejection of claims 16, 17, 19 and 38.

B. <u>35 U.S.C. § 103, Alleged Obviousness, Claim 20</u>

The examiner has rejected claim 20 under 35 U.S.C. § 103(a) as being unpatentable over Brendel in view of Coile et al. (U.S. Patent No. 6,061,349). This rejection is respectfully traversed for at least the same reasons as set forth above with regard to independent claim 16 from which claim 20 depends. That is, Brendel does not teach a dispatch layer between a TCP layer and an IP layer or such a dispatch layer that has the four modes of operation recited in claim 16. Coile, likewise, does not teach or suggest these features.

Coile is cited merely as teaching server daemon processes. Coile does not cure the deficiencies of Brendel and thus, any alleged combination of Brendel and Coile, even if such a combination were possible and one of ordinary skill in the art were somehow motivated to attempt such a combination, would not result in the invention as recited in independent claim 16, from which claim 20 depends.

Therefore, appellants respectfully submit that neither Brendel nor Coile, either alone or in combination, teach or suggest all of the features of claim 20. Accordingly, appellants respectfully request that the Board reverse the examiner's Final Rejection of claim 20.

CONCLUSION

For all the above reasons, appellants submit that the Final Rejection of claims 16, 17, 19, 20 and 38 is improper, and respectfully request that the Final Rejection be reversed.

Gerald H. Glanzman
Registration No. 25,035
YEE & ASSOCIATES, P.C.
PO Box 802333
Dallas, TX 75380
(972) 385-8777

GHG/bj

CLAIMS APPENDIX

The text of the claims involved in the appeal are:

16. A computer comprising:

a plurality of processes, wherein the plurality of processes service a destination address and have process addresses;

a packet routing layer, wherein the packet routing layer routes packets to the plurality of processes using a destination addresses within the packets;

a dispatch layer between a TCP layer and an IP layer, wherein the dispatch layer has a plurality of modes of operation including:

a first mode of operation in which the dispatch layer receives a packet from a client, wherein the packet includes the destination address;

a second mode of operation, responsive to receiving the packet, in which the dispatch layer identifies a process within the plurality of processes to service the client, wherein the process is an identified process;

a third mode of operation in which the dispatch layer translates the destination address to a process address for the identified process within the plurality of processes; and

a fourth mode of operation, responsive to the third mode of operation, in which the packet is sent to the packet routing layer.

17. The computer of claim 16, wherein each packet includes a source address and wherein the dispatch layer further includes:

a fifth mode of operation in which the dispatch layer receives a packet from the identified process for the client; and

a sixth mode of operation, responsive to the fifth mode of operation, in which the dispatch layer translates the source address in the packet of the destination address.

19. The computer of claim 16, wherein the packet routing layer is a transmission control protocol layer.

20. The computer of claim 16, wherein the plurality of processes is a plurality of server daemons.

38. A computer program product for routing packets from a client to a selected process within a plurality of processes servicing a connection between the data processing system and the client comprising:

a computer readable medium;

first instructions for receiving a packet for the connection between the data processing system and the client, wherein the packet includes a destination address; and

second instructions for translating, in a dispatch layer between a TCP layer and an IP layer, the destination address to an intermediate destination address, which is an address for the selected process within the plurality of processes, wherein the instructions are embodied within the computer readable medium.

EVIDENCE APPENDIX

There is no evidence to be presented.

RELATED PROCEEDINGS APPENDIX

There are no related proceedings.